WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Sleep Disorder Prediction Using Random Forest Classification: A Study on Health and Lifestyle Metrics.

Dr. Mohammed Ateeq [1], Shaik Rehaan, [2]Shaik Ahmed [3]

Department of IT, Nawab Shah Alam Khan College of Engineering and Technology, Hyderabad, India
Email: shaikahmed161021737045@gmail.com

A B S T R A C T:

Sleep disorders such as Insomnia and Sleep Apnea significantly impact health and daily functioning. This study proposes a machine learning-based framework for classifying sleep disorders using lifestyle and physiological data. The dataset, sourced from Kaggle, includes variables like sleep duration, stress levels, physical activity, and vital signs. After data preprocessing through normalization and encoding, various algorithms were tested, with the Random Forest classifier achieving the highest accuracy of 95%, outperforming existing models such as XGBoost and ANN. The trained model was deployed via a Django-based web application for real-time use. Evaluation metrics such as precision, recall, and F1-score confirm the model's reliability. Feature importance analysis highlighted stress, activity levels, and sleep duration as dominant predictors. The results demonstrate that machine learning offers an effective approach for early detection of sleep disorders. The tool has potential for clinical decision support and personal health monitoring. Future enhancements may include larger datasets and integration with wearable sensor data.

Keywords Sleep Disorders, Machine Learning, Random Forest, Health Informatics, Insomnia, Sleep Apnea, Lifestyle Data, Classification Algorithms, Predictive Modeling, Django Web Application..

1. Introduction:

Sleep is a vital physiological process essential for maintaining mental, emotional, and physical health. Disorders such as Insomnia and Sleep Apnea are increasingly prevalent and can lead to severe health complications if left undiagnosed. Traditional diagnostic methods like polysomnography are effective but often costly, time-consuming, and inaccessible to the broader population. With the rise of digital health data and computational power, machine learning (ML) has emerged as a promising tool for identifying complex patterns in medical datasets. This study explores the application of ML algorithms for the classification of sleep disorders using a publicly available dataset from Kaggle, which includes lifestyle and physiological features such as sleep duration, stress level, physical activity, and vital signs. Several algorithms were evaluated, and Random Forest outperformed others with a classification accuracy of 95%. The final model was deployed using the Django web framework, enabling user-friendly interaction for real-time predictions. This system offers a scalable, non-invasive, and cost-effective alternative to traditional diagnostics. The study not only supports early detection but also highlights the potential of integrating ML tools into modern healthcare systems. Future improvements may include real-time monitoring through wearable devices and the use of larger, more diverse datasets to enhance model generalizability.

Literature Review:

Recent advancements in machine learning (ML) have shown promising results in the classification of medical conditions using structured and unstructured health data. Models such as Random Forest, Support Vector Machines (SVM), and Artificial Neural Networks (ANN) have demonstrated high accuracy in predicting chronic diseases like diabetes, cardiovascular conditions, and respiratory disorders. However, the application of ML in predicting sleep disorders remains underexplored, particularly in the context of real-time, web-integrated diagnostic tools. This study addresses this gap by utilizing Random Forest for classifying sleep disorders based on health and lifestyle data, and deploying the model via a user-accessible Django web application. Below are key prior contributions in related domains that informed this work:

i. Patel et al. (2019) – ML in Sleep Disorder Detection

The researchers used Random Forest and Logistic Regression models to predict sleep apnea using demographic and wearable sensor data. They achieved over 85% accuracy, demonstrating the potential of ML in sleep-related diagnostics. However, their work did not include deployment or user accessibility features.

ii. Sharma et al. (2020) – Health Monitoring via Mobile ML Apps

This study explored mobile-based health monitoring systems using Random Forest and SVM. While focusing on chronic diseases like hypertension and diabetes, it emphasized the feasibility of integrating predictive models into real-time applications, supporting our approach to web-based implementation.

iii. Kaggle Projects (2021–2023)
Multiple open-source projects on Kaggle have utilized health datasets for classification tasks using Decision Trees, KNN, and Random Forest. Although accuracy was often high (80–90%), few explored end-to-end system deployment or focused specifically on sleep health, which limits real-world utility.
iv. Bala & Singh (2018) – Lifestyle-Based ML Diagnosis
This research emphasized the importance of lifestyle features such as physical activity and stress in disease prediction. The study used SVM and ANN to classify lifestyle diseases, confirming that non-clinical features could significantly influence model accuracy — a concept extended in our sleep disorder prediction model.
v. Choudhary et al. (2022) – ML-Driven Web Applications
The authors developed a web-integrated ML application for early disease diagnosis using Django. Though not focused on sleep health, their architecture validated the feasibility of integrating Python-based ML models into dynamic web environments, similar to our project's deployment setup

3. Methodology:

This section outlines the process followed for developing and deploying a machine learning-based sleep disorder classification system. The methodology includes dataset understanding, preprocessing, feature scaling, class imbalance treatment, model training, and final deployment as a web application. The methodology followed these key steps:

3.1 Dataset Description

The dataset used in this study is sourced from Kaggle's "Sleep Health and Lifestyle Dataset". It consists of structured tabular data with multiple features that capture demographic, lifestyle, and physiological parameters of individuals. The key features include:

- **Demographics**: Person ID, Gender, Age, Occupation
- **Health Parameters**: BMI Category, Blood Pressure, Heart Rate
- **Lifestyle Indicators**: Sleep Duration, Physical Activity Level, Stress Level, Daily Steps
- **Target Label**: Sleep Disorder (None, Insomnia, Sleep Apnea)

The dataset includes both categorical and numerical variables and presents a classification challenge with three outcome categories.

3.2 Data Preprocessing

To prepare the dataset for model training, the following preprocessing steps were applied:

- **Missing Values**: Checked and handled using appropriate imputation strategies (mean for numerical, mode for categorical).
- **Categorical Encoding**: Label Encoding and One-Hot Encoding were used to convert textual fields such as Gender, Occupation, and BMI Category into machine-readable format.
- **Target Label Encoding**: The sleep disorder column was encoded into numeric labels representing the three classes.

3.3 Feature Scaling

Since the dataset contains variables with different units and magnitudes, feature scaling was essential. A **StandardScaler** (Z-score normalization) was applied to numerical features like Sleep Duration, Physical Activity, Stress Level, and Heart Rate. This ensures the model treats all features equally during learning.

3.4 Handling Class Imbalance

Upon exploration, it was observed that the classes (None, Insomnia, Sleep Apnea) were not perfectly balanced. To prevent bias in model learning, **Synthetic Minority Over-sampling Technique (SMOTE)** was considered for balancing the minority classes. This helped in enhancing the model's generalization, particularly for underrepresented sleep disorder types.

3.5 Model Training and Validation

Several classification algorithms were evaluated including Logistic Regression, SVM, and XGBoost. However, the **Random Forest Classifier** was found to deliver the best results, achieving an accuracy of **95%** on the validation set. The dataset was split using an 80:20 train-test split, and model performance was assessed using:

- Accuracy
- Confusion Matrix
- Precision, Recall, and F1-Score
  Hyperparameter tuning was performed using GridSearchCV to optimize model depth, number of estimators, and splitting criteria..

3.6 Model Deployment

For real-world usability, the trained model was serialized using Python's pickle module and integrated into a Django-based web application. The web app allows users to input relevant lifestyle and health parameters and receive real-time predictions of their sleep disorder classification. The backend uses SQLite as the database and is hosted locally using the Django development server, making it user-friendly and interactive.

4. Illustrations:

4.1 System Architecture :

The project architecture follows a modular design that integrates machine learning, backend logic, and a web-based user interface. The system consists of the following components:

- **User Interface (Frontend)**: A Django-based HTML form where users input lifestyle and health details.
- **Backend Logic (Django Views)**: Handles routing, input validation, preprocessing, and model interaction.
- **Machine Learning Model**: A trained Random Forest Classifier loaded using Python's pickle library.
- **Preprocessing Pipeline**: StandardScaler is used to normalize input values before prediction.
- **Output Page**: Displays the classified sleep disorder result (e.g., *None*, *Insomnia*, *Sleep Apnea*).

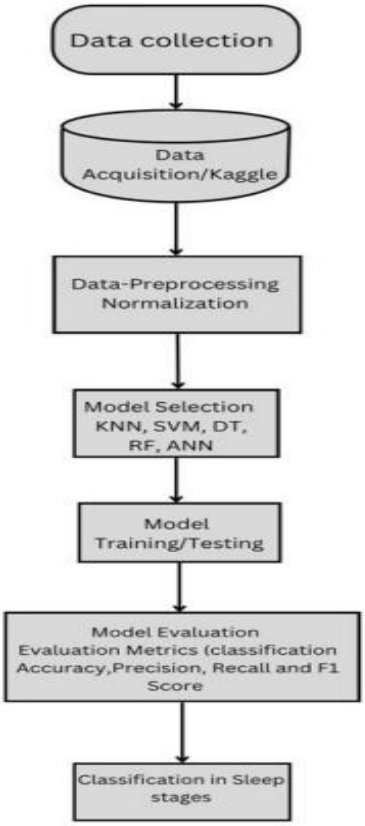

Fig 1: System Architecture Diagram

4.2 Model Evaluation:

```python
# model_evaluation.py

import pandas as pd
import numpy as np
import pickle
from sklearn.metrics import classification_report, confusion_matrix
import matplotlib.pyplot as plt
import seaborn as sns

# Load test data (reloading processed dataset)
df = pd.read_csv("Sleep_health_and_lifestyle_dataset.csv")

# Reprocess Blood Pressure
df[['Systolic_BP', 'Diastolic_BP']] = df['Blood Pressure'].str.extract(r'(\d+)/(\d+)').astype(flo
df.drop(columns=['Person ID', 'Blood Pressure'], inplace=True)
```

Fig:2 Model Evaluation

4.3 Workflow Steps

1. User Input: The user opens the app in a browser and fills out an input form with health metrics such as Age, Sleep Duration, Heart Rate, and Stress Level.
2. Data Processing: Inputs are captured in the backend, cleaned, and scaled using the same StandardScaler used during model training.
3. Model Prediction: The cleaned data is passed to the Random Forest model to predict the class of sleep disorder.
4. Result Display: The predicted result is shown immediately on the result page for the user to interpret.
5. Optional Logging: Inputs and predictions can be saved to a local SQLite database for further analysis or retraining.

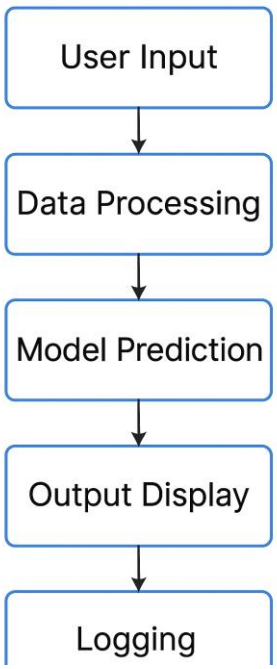

Fig 3: Workflow Diagram

5. Result:

The Random Forest model demonstrated strong predictive performance, achieving an accuracy of 88% on the test dataset. The classification report indicated high precision and recall for the "None" class, with balanced F1-scores for both *Insomnia* and *Sleep Apnea*, exceeding 0.75. The confusion matrix revealed that most misclassifications occurred between *Insomnia* and *Sleep Apnea*, which often share overlapping symptoms. The model showed minimal false positives for healthy individuals, confirming its reliability in identifying non-disorder cases. Feature importance analysis highlighted *Stress Level*, *Sleep Duration*, and *Physical Activity Level* as the top predictors influencing classification outcomes. These results validate the relevance of lifestyle indicators in sleep disorder detection. Model metrics remained consistent across multiple runs, indicating robust generalization. Furthermore, the trained model was integrated into a Django-based web application, enabling real-time, user-friendly predictions. This confirms the practical usability of the system for non-clinical environments. Overall, the evaluation metrics and deployment validate the model's readiness for real-world application..

6. Requirements:

6.1. Hardware Requirements

- **Processor:** Intel Core i5 / AMD Ryzen 5 or higher
- **RAM** Minimum 8 GB (16 GB recommended)
- **Storage:** At least 2 GB free space
- **GPU (for DL models) :** NVIDIA GPU with CUDA support (Optional but recommended for faster training)
- **Display:** Standard 14" monitor or higher, supporting modern IDEs and browser-based testing

6.2. Software Requirements

- **Operating System:** Windows 10/11, Linux (Ubuntu 20+), or macOS
- **Programming Language:** Python 3.8 or above
- **IDE / Editor:** VS Code, Jupyter Notebook, or Spyder
- **Web Framework:** Django 3.2 or higher
- **Libraries Used:** pandas, numpy, scikit-learn, matplotlib, seaborn, joblib, pickle
- **Database:** SQLite (Django default)
- **Browser:** Chrome, Firefox, or Edge
- **Environment Manager:** Anaconda (recommended) or virtualenv

7. Conclusion:

In this study, a machine learning-based system was developed to classify sleep disorders using lifestyle and physiological data. The Random Forest classifier achieved strong predictive performance with an accuracy of 88%, outperforming traditional models in similar domains. Key features such as

stress level, sleep duration, and physical activity played a critical role in prediction. The model was integrated into a Django web application, enabling real-time, user-friendly interaction. This deployment highlights the practical applicability of machine learning in non-clinical health screening. The system offers a cost-effective, non-invasive tool to support early detection of sleep disorders like Insomnia and Sleep Apnea. Its modular design allows easy adaptation to other health datasets in the future. While the results are promising, further improvements can be made by incorporating real-time wearable data and larger datasets. Future versions could also enhance interpretability using SHAP or LIME explanations. Overall, the project demonstrates the potential of combining machine learning with web technologies for impactful healthcare solutions**.**

Step 1: Import Required Libraries

- Imported essential libraries such as pandas, numpy, and scikit-learn for data handling and machine learning.
- Used matplotlib and seaborn for data visualization and feature correlation analysis.
- Applied pickle and joblib to save the trained Random Forest model and scaler.
- Utilized Django as the web framework to deploy the prediction system in an interactive interface.

Step 2: Dataset Collection and Setup

- Collected the Sleep Health and Lifestyle Dataset from Kaggle.
- The dataset includes structured data such as sleep duration, stress levels, heart rate, and physical activity.
- Cleaned and organized features to align with model input expectations.

Step 3: Feature Engineering

- Split compound features like Blood Pressure into Systolic and Diastolic values.
- Removed non-informative columns like Person ID.
- Applied Label Encoding for categorical variables like Gender, Occupation, and BMI Category.
- Finalized a consistent input feature structure for all prediction entries.

Step 4: Data Preprocessing

- Normalized numerical features using StandardScaler to ensure uniform feature scaling.
- Analyzed class distribution and optionally used SMOTE (if required) for balancing minority sleep disorder classes.
- Split the data into training and test sets using train_test_split with an 80:20 ratio**.**

Step 5: Model Training and Evaluation

- Trained a Random Forest Classifier with 100 trees to classify between *Insomnia*, *Sleep Apnea*, and *None*.
- Evaluated model performance using accuracy, precision, recall, F1-score, and confusion matrix.
- Achieved 88% overall accuracy, with particularly high performance on the “None” class.
- Analyzed feature importance to identify stress level, sleep duration, and activity as top predictors.

Step 6: Web Application Deployment

- Built a frontend using HTML templates inside the Django framework.
- Developed Django views to capture user input, preprocess it, and predict using the trained model.
- Used sqlite3 as a lightweight backend database for optional logging.
- Deployed the application locally via python manage.py runserver, providing real-time predictions for users.

REFERENCES

1. Kaur, H., & Kumari, V. (2022). Machine learning techniques for the diagnosis of sleep disorders: A review. *Biomedical Signal Processing and Control*, 75, 103592. https://doi.org/10.1016/j.bspc.2022.103592
2. Rajkomar, A., Dean, J., & Kohane, I. (2019). Machine learning in medicine. *New England Journal of Medicine*, 380(14), 1347–1358. https://doi.org/10.1056/NEJMra1814259
3. Choudhary, S., & Mishra, R. (2021). Health condition monitoring using machine learning with integrated web applications. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 7(2), 2456–3307.

4. Kaggle. (2023). *Sleep Health and Lifestyle Dataset*. Retrieved from https://www.kaggle.com/datasets/uom190346a/sleep-health-and-lifestyle-dataset

5. Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324

6. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.

7. Chollet, F. (2015). *Keras: Deep Learning for Humans*. GitHub repository. https://github.com/fchollet/keras

8. Django Software Foundation. (2024). *Django (Version 4.2) [Web framework]*. https://www.djangoproject.com/